

New Algorithm to Drawing a Random Polygon from a Set of Flat Points

A. Nourollah

M. Abooei Mehrizi

Faculty of Electrical & Computer Engineering of Shahid Rajaee Teacher Training University, Tehran, Iran

Abstract

Generating random polygons problem is important for verification of geometric algorithms. Moreover, this problem has applications in computing and verification of time complexity for computational geometry algorithms. Since it is often not possible to get real data, a set of random data is a good alternative. In this paper, a heuristic algorithm is proposed for generating random polygons that is using convex hulls to generate the random polygon.

Keywords: *Convex hull, Random Polygon, Visibility, Computer Graphics.*

1. Introduction

The generation of random geometrical objects has received some attention by researchers [2][3][5]. A challenge of these problems is the generation of random simple polygons. Since no polynomial time algorithm is known to solve the problem, researchers either try to use heuristic algorithms which don't have uniformed distribution or restrict the problem to certain classes of polygon such as monotone and star-shaped polygons [1][3][4].

The importance of geometric objects application is the simplicity of testing geometric algorithms. Since a set of data may become both too large and too hard to define for practical purposes, what one might do is to use randomly generated data that has a high probability to cover all the different classes of inputs. Thus, since practical data may not be available for testing, it is natural to test the algorithm on randomly input data.

Polygons are one of the fundamental building blocks in geometric modeling and they are used to present a wide variety of shapes and figures in computer graphics, vision, pattern recognition, robotics and other computational fields.

Some recent applications address uniformed random generation of simple polygons with given vertices, in the sense that a polygon will be generated with probability if there exist a total of $1/T$ simple polygons with such vertices.

The following sections of this paper have been organized in this way: section 2 has been allocated to related works. In section 3 the needed preliminary concepts are stated. In section 4 the proposed algorithm is posed for generating random polygon, in section 5 its performance and accuracy are investigated and finally in section 6 the conclusion will be discussed.

2. The Related Works

Recently, the generation of random geometric objects and especially simple polygons has received some attention by researchers. For example, Epstein studied the uniformly random generation of triangulations [6]. Zhu et al. presented on an algorithm for generating x -monotone polygons on a given set of vertices uniform at random [3]. A heuristic for the generation of simple polygons was investigated by O'Rourke and Virmani [5]. Auer and Held presented the following heuristic algorithms:

- Steady Growth: an incremental algorithm adding one point after the other whose time complexity in the worst case and in the best case is $O(n^2)$ and $O(n \log n)$ respectively.
- Space e Partitioning: which is a divide and conquer algorithm and it has s time complexity of $O(n^2)$.
- Permute & Reject: which creates random permutations (polygons) and surveys whether it is corresponding with a simple polygon or not, until a simple polygon is encountered. Its complexity is $O(n \log n)$.
- 2-opt Moves: which by starting from a completely random polygon and replacing its intersected edges encounters a simple polygon and its complexity is $O(n^4)$ [2].

3. Preliminaries

Let S be a set of random vertices, there exists T simple polygons on S in total, such that every polygon is generated with probability $1/T$. It is supposed that no three points are linear. A simple polygon is a limited plane by a limited set of line segments that form a simple closed curve. In other words, a simple polygon P on S , is a polygon whose edges don't intersect one another except on vertices S .

A convex polygon, is a simple polygon which for both of vertices x and y from polygon, the line segment $\overline{xy}$ lies inside P or on its border, i.e., $\overline{xy} \subseteq P$. The convex hull of a finite set of points on the plane ($CH(S)$) is the smallest convex polygon P that encloses S . The smallest polygon means that there is no polygon P' such that $S \subseteq P' \subset P$ (Fig.1).

Let k be the number of all convex hull layers on the set of points S . Every layer is defined by $l_c (k >$

1 and $1 \leq l_c \leq k$). By the supposition of numbering layers from the most internal to the most external convex hull layers, n_i is supposed to be the number of the most internal convex hull layer, i.e., l_1 .

Vertex x sees vertex y if $\overline{xy} \subseteq P$ and $\overline{xy}$ doesn't lie out of P . In this case y is visible from x or in other words x has visibility toward y . The polar sorting, is the sorting of a set of vertices around a given vertex according to polar angle. The star-shaped polygon, is a polygon which is visible at least from an interior vertex. In the next section, the proposed algorithm is posed to generate random polygon.

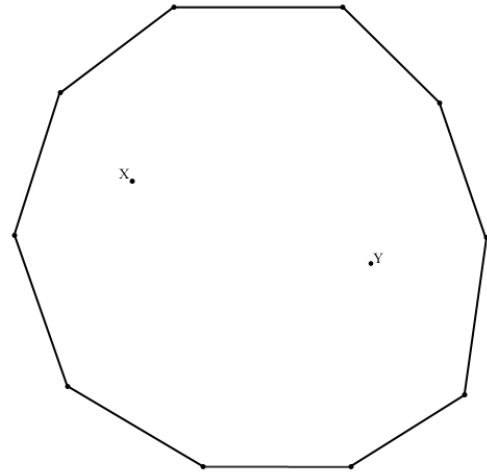


Fig.1. Convex hull, x has visibility toward y .

4. The proposed Algorithm

In this section, the algorithm is posed to generate the random polygon with time complexity $O(n \log n)$ in which n is the number of vertices.

To generate a random polygon on a vertex set it is performed in this way that first, draw a random vertical line between leftmost and rightmost points (as $l1$), then draw the first convex hull (as $ch1$) such that the $l1$ just have one intersect with the $ch1$, then draw the other convex hulls with especial condition. Some different algorithms are proposed [7][8][9] to generate convex hulls. Chazelle had prepared an algorithm in [10] with complicity of $O(n \log n)$ for generate convex hulls, means; it is optimal in time complicity. At last, draw a

line between two reminded points; the polygon is generated. In below the algorithm is presented in step by step.

First step: draw random imaginary line

In this step, should find leftmost and rightmost vertices, and then draw a vertical random line between leftmost and rightmost vertices as $l1$. The final shape of polygon dependent to position of the $l1$ will be different, and with changing the position of the $l1$ in a set of fixed points, the final polygon shape will change.

Second step: generate convex hulls

Draw first convex hull such that just one line of two, that have intersect with the $l1$, is drawing and the other should not (Fig.2).

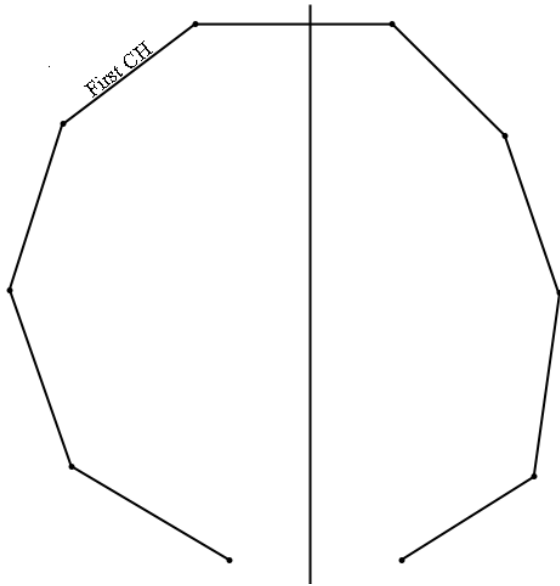


Fig.2. generate first convex hull and do not draw one of two intersected lines.

Then, all of the vertices with two degree should remove from the set of vertices and draw internal convex hulls in this way:

Draw convex hull of reminded vertices such that nowhere have intersecting with the $l1$, then remove two degree vertices from the set and do this until stay just one point in each side of line; at the end connect the vertices (two reminded vertices) together, polygon is completed (Fig.3).

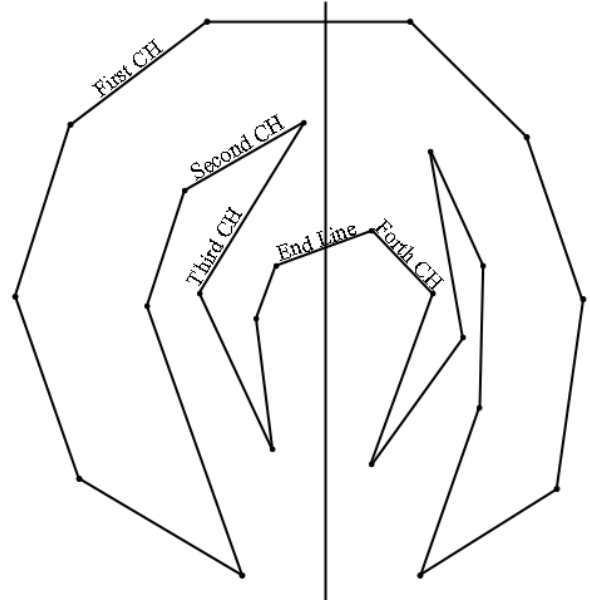


Fig.3. generate internal convex hulls such that do not intersect $l1$

It goes without saying, the vertices that are placed on the $l1$, without changing the position of them will be classified in one of two classes (one class is contained of vertices that are in left side of $l1$, and the other class is contained of vertices in right side of $l1$).

Below is shown the proposed algorithm.

Algorithm: polygon generation using Convex Layers

Begin

Step1:

Label all of the points as "**free**".

Draw a random vertical line between leftmost and rightmost points; label it as "**L1**".

Step2:

Draw convex layer except one of two edges where crossed L1.

Label the points as "**used**" where two links with others have.

While ($|\text{Free Points}| > 2$)

Begin

Draw convex layer where no edge cross L1.

Label the points as "**used**" where two links with others have.

End While

Draw an edge between two free points.

End

5. Verification of algorithm performance

In this section, the performance of proposed algorithm is investigated by computing its time complexity. In this algorithm, time of finding leftmost and rightmost points is $O(n)$ and drawing the convex layers is $O(n \log n)$; then complexity of this algorithm is $O(n \log n)$, it means, this algorithm is optimal in performance.

6. Conclusion

In this paper, we proposed a new simple algorithm to draw a simple and random polygon from a set of random flat points. In this algorithm, first draw a line between leftmost and rightmost points, that with changing the position of this line in x axis will arrive to different polygons. In next step will draw convex hulls and remove edges that have intersected the line, and do this for points that have less than two connections with others (and connecting last two points) will arrive to final random simple polygon. Time complexity of this algorithm is $O(n \log n)$; It means this algorithm is optimal in execution.

In this paper we imagined the position of points are prepared before running the algorithm; later we will suggest an algorithm such that generates a random polygon when points are given when algorithm is running and the points are reducing dynamically.

7. References

- [1] McDonald, B., Smith, I. "Partial symmetry breaking"; In: P. Van Hentenryck(Ed.), Proc. Of CP'02 LNCS2470, Springer-verlag, 431-445, 2002.
- [2] Auer, T., Held, M. "Heuristics for the Generation of Random Polygons"; Proc. 8th Canadian on Computational Geometry (CCCG'96), 38-41, 1996.
- [3] Zhu, C., Sundaram, G., Soneyink, J., Mitchell, J.S.B. "Generating Random Polygon with Given Vertices"; Comput. Geom. Theory and Appl., Vol 6, Issues 5, 277-290, 1996.
- [4] Sohler, C. "Generating random star-shaped polygons"; In: Proc. 11th Canadian Conference on Computational Geometry (CCCG'99), 174-177, 1999.
- [5] O'rourke, J., Virmani, M. "Generating Random Polygons"; Technical Report 011, CS Dept. Smith Colege, Northhampton, MA 01063 1991.
- [6] Michael, T.S. "How to Guard an Art Gallery and other Discrete Mathematical Adventures"; the Book of the Johns Hobkins University Press, printed in the United States of American, 2009.
- [7] Green P. J., Silverman B. W., Constructing the convex hull of a set points in the plane, Journal of Computer, Vol. 22, pp. 262-266, 1977.
- [8] Overmars M. H., Leewen J., Maintenance of configuration in the plane, computing system and science, Vol. 23, pp. 166-204, 1981.
- [9] Toussaint G. T., Pattern recognition and geometrical complexity, 5th International Conference of Pattern Recognition, pp. 1324-1347, 1980.
- [10] Chazelle B., On the Convex Layers of a Planar Set, IEEE Tran. Information Theory, Vol. IT-31, No. 4, pp. 509-517, 1985.